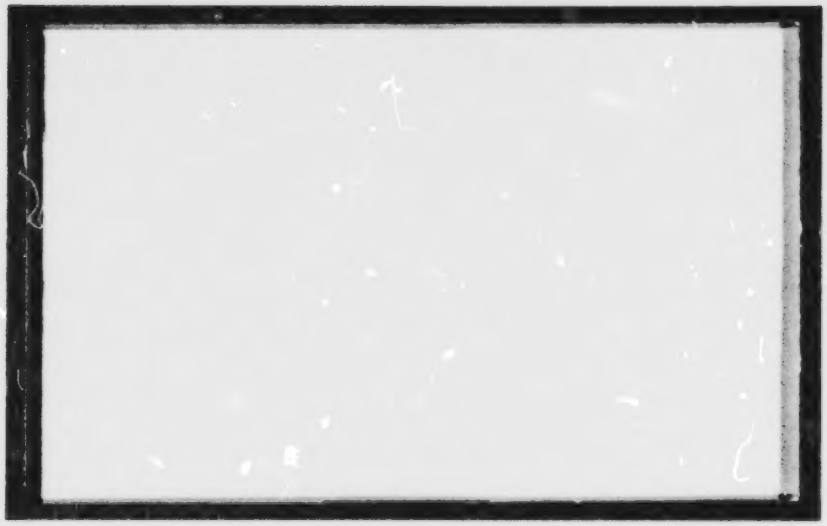
UNIVERSITÉ DE MONTRÉAL



CENTRE DE RECHERCHE SUR LES TRANSPORTS

CASE POSTALE 6128, SUCCURSALE A, MONTRÉAL, QUÉBEC, H3C 3J7

EMME

COMMAND LANGUAGE REFERENCE MANUAL

by

Linda James-Lefebvre and Claude Achim

This project was developed with funding provided under the terms of a research contract with Transport Canada

Université de Montréal

Centre de recherche sur les transports - Publication # 128

Mars 1979

Dépôt légal - 2e trimestre 1979

E M M E

(Equilibre Multimodal / Multimodal Equilibrium)

The EMME system is a set of computer programs designed for two-mode transportation planning. It is based on an integrated two-mode traffic equilibrium method which, roughly speaking, combines a zonal aggregate demand model with an equilibrium type road assignment and a transit assignment method. The model is described in "EMME: A Planning Method for Multimodal Urban Transportation Systems", by Florian *et al.*, Publication # 62, Centre de recherche sur les transports, Université de Montréal.

The purpose of this manual is to give a description of the command language through which the user specifies the operations he wishes to perform. Details about the definition and coding of the data required by EMME and other informations concerning the use of EMME can be found in the "EMME User's Guide".

Ce manuel est également disponible en français.

TABLE OF CONTENTS

1. Introduction. 1

2. Command and subcommand syntax 2

3. Commands. 3

4. $ASSIGN Perform an assignment. 4

4.1. $ASSIGN options. 4

4.2. $ASSIGN subcommands. 6

4.2.1. IDENT set computer file identification. 6

4.2.2. PARAMETERS set assignment parameters. 7

4.2.3. $ASSIGN parameter descriptions. 9

4.3. Sample assignment requests 19

4.3.1. Bimodal assignment. 19

4.3.2. Bimodal assignment, minimum specification 21

4.3.3. Road traffic assignment with fixed demand 22

4.3.4. Road traffic assignment with variable demand. 23

4.3.5. Transit assignment with fixed demand. 24

4.3.6. Transit assignment with variable demand 25

5. $COMMENT Print a comment line 26

6. $DATABANK Perform databank operations 27

6.1. $DATABANK option 27

6.2. $DATABANK subcommands. 28

6.2.1. CORRECTION Request addition, modification or deletion of data. 29

6.2.1.1. CORRECTION options 32

6.2.1.2. Headers and data records 32

6.2.1.3. INCLUDE records. 35

6.2.1.4. Header SCALE 35

6.2.1.5. Header O-D MATRIX. 36

6.2.1.6. END record 36

6.2.2. SCENARIO Enter scenario data into the databank 37

6.2.2.1. SCENARIO options 37

6.2.3. USE Use a scenario 39
6.2.3.1. USE options. 39
6.2.4. PRINT Print data from the databank 40
6.2.4.1. PRINT header
Print data of a specified type 40
6.2.4.2. PRINT SCENARIO name
Print scenario data. 40
6.2.4.3. PRINT SCENARIO LIST
Print list of scenario names 41
6.2.5. PUNCH Punch data from the databank 42
6.2.5.1. PUNCH header
Punch data of a specified type 42
6.2.5.2. PUNCH SCENARIO name
Punch scenario data. 43
6.2.6. PURGE Delete data from the databank. 44
6.2.6.1. PURGE header
Delete all data of a specified section . 44
6.2.6.2. PURGE SCENARIO name
Delete scenario data 44
7. $DEBUG Control debugging output 45
7.1. $DEBUG options 45
8. $END Terminate EMME 45
8.1. $END option. 46
9. $REPORT Print reports. 47
9.1. $REPORT options. 47
9.2. $REPORT subcommands. 48
9.2.1. IDENT Set computer file identification 48
9.2.2. PARAMETERS Set assignment parameters 48
9.2.3. LINK Print road link report. 50
9.2.3.1. LINK parameter description 50
9.2.4. DEMAND Print predicted auto demand report. 53
9.2.4. AUTO TIME DISTRIBUTION
Print automobile trip time distribution. 55
10. $TITLE Enter title line text 56
Appendix A: EMME Coding forms. 57
Appendix B: EMME Command, option and subcommand summary. 71

1. Introduction

The EMME user may specify the operations he wishes to perform through a simple language which consists of commands and subcommands. For example to perform a bimodal assignment the user might specify

```
$TITLE,1,WINNIPEG 1976
$TITLE,2,PORTAGE PARKING CURTAILED
$ASSIGN,BIMODAL
IDENT NETWORK 1976, PORTAGE PARKING
PARAMETERS SPLT=.5 AOCC=1.3 RTIM=500
PARAMETERS ITRD=30
PARAMETERS WMAX=30
$REPORT
LINK
DEMAND
$END
```

This example gives the information for the 2 title lines on all reports in the first two commands, then requests a bimodal assignment ($ASSIGN,BIMODAL). IDENT gives the identification text to be included on all the computer output files in order to more easily identify a given file. The 3 PARAMETERS subcommands specify assignment parameters which the user chose to specify:

- modal split = .5
- auto occupancy = 1.3 persons
- computer time limit for road traffic assignment = 500 sec.
- road traffic assignment iteration limit = 30 iterations
- maximum waiting time (transit) = 30 minutes

$REPORT requests reports on the assignment. LINK requests the road link report and DEMAND requests the predicted demand report. $END terminates the command sequence.

2. Command and subcommand syntax

In the syntax notation { } indicates that one of the items between the braces must be chosen, [ ] indicates that one or more of the items between the parentheses must be chosen. ␣ represents a blank or space.

Commands have the following form

<command code> $\{\begin{matrix}␣\\,\end{matrix}\}$ <option> $\cdot \{\begin{matrix}␣\\,\end{matrix}\}$ <option> ...

that is a command code followed by one or more blanks or commas, which may be followed by options separated by one or more blanks or commas. The last option (or the command code if there are no options) must be followed by a blank (not a period). The first character of a command is always $.

An example of a command is

$ASSIGN,ROAD,FIXED

where $ASSIGN is the command code and ROAD and FIXED are options.

Subcommands have the following general format

<subcommand code> $\{\begin{matrix}␣\\,\end{matrix}\}$ <other information>

that is a subcommand code, one or more blanks or commas, followed by other information which is different for different subcommands. The first character of a subcommand code is never $. The syntax of each individual subcommand is described in the section on the subcommand.

Command codes and subcommand codes may be abbreviated, however letters may not be omitted. At least 3 characters must be used. For example the command $ASSIGN,... may be written alternatively as

$ASS,...

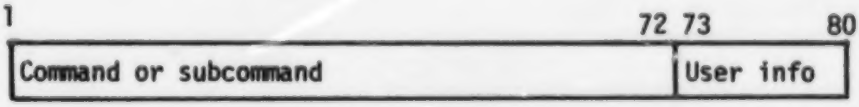
or $AS,... ($AS is minimum 3 characters)

$ASSGN,... is not valid since the I is omitted.

These abbreviation rules do not apply to $DATABANK and its subcommands (see section 6).

Blanks may appear before the command or subcommand codes.

Commands and subcommands have a maximum length of 72 characters which are followed by 8 characters of user information. $DATABANK and its subcommands use a complete card (80 characters) and do not consider a user info field.



3. Commands

EMME currently accepts the following commands. Other commands, needed to control additional capabilities in the system, can be added easily

$ASSIGN	Perform an assignment
$COMMENT	Print comment line
$DATABANK	Perform databank operations
$DEBUG	Control debugging output
$END	Terminate EMME
$REPORT	Print reports
$TITLE	Enter title information

Descriptions of their options and subcommands follow.

4. $ASSIGN Perform an assignment

Command format:
$ASSIGN,<option>,<option>

$ASSIGN requests an assignment. The type of assignment is specified in the command options. The assignment can be controlled by parameters entered in the PARAMETERS subcommand. The IDENT subcommand may be used to enter a unique identification for computer files produced by the assignment.

4.1 $ASSIGN options

The options recognized are

Option	Type of assignment
BIMODAL	Bimodal
ROAD	Road traffic only
TRANSIT	Transit only
FIXED	Fixed demand
VARIABLE	Variable demand

Valid combinations of these options i.e. assignments which EMME can currently perform are

Options	Assignment
BIMODAL	Bimodal
ROAD,FIXED	Road traffic only with fixed demand
ROAD,VARIABLE	Road traffic only with variable demand
TRANSIT,FIXED	Transit only with fixed demand
TRANSIT,VARIABLE	Transit only with variable demand

Options may be abbreviated, but letters may not be omitted. A minimum of the first 3 letters is required. For example TRANSIT,VARIABLE might be abbreviated to

TRANS,VAR or TRA,VAR

but <u>not</u> to TRNS,VARA (A omitted from TRANS, I omitted from VARIA)

4.2 $ASSIGN subcommands

The following subcommands are recognized

IDENT

PARAMETERS

Neither subcommand is required. The assignment procedure will use default values in place of the information which the user may specify in these subcommands

4.2.1. IDENT Set computer file identification

The subcommand has the following form:

IDENT {b,}<identification text>

that is the subcommand code IDENT, one or more blanks or commas, followed by up to 40 characters of text (including embedded blanks). The identification would normally indicate the particular characteristics of the assignment to be performed. The identification text is inserted in the first record of the computer files produced during the assignment process, to aid the user in easily identifying files produced from several different assignment. Example of an IDENT subcommand:

IDENT NETWORK 1976, PORTAGE PARKING

If the IDENT subcommand is omitted the default identification

EMME ASSIGNMENT

will be used.

4.2.2 PARAMETERS Set assignment parameters

This subcommand has the following form

PARAMETERS {b,} <keyword = value> b <keyword = value>

that is the subcommand code, one or more blanks or commas, followed by 'keyword = value' pairs separated by one or more blanks.

A keyword is a 4 character identifier for a parameter which the user wishes to set to the value which follows the equals sign (required). For example:

AOCC=1.3

where AOCC is the keyword for the auto occupancy parameter which the user wishes to set to the global value 1.3 persons per auto.

Some parameters permit character values, for example:

BUSF=COMP

where BUSF is the keyword for the bus frequency parameter and COMP indicates that the bus frequency should be calculated and used in the road traffic assignment.

One parameter, HDWY, permits multiple values separated by commas.

Keywords and character values may not be abbreviated.

A numeric parameter value may contain the following:

- maximum 7 digits (including explicit zeros)
- preceding plus or minus sign
- decimal point

In the following parameter value description, x indicates a digit, '.' indicates the usual position of the decimal point. The format of numeric values, shown for example as x.xx, gives the usual form of the value. In this example the value would have 1 digit before and 2 digits after the decimal point. The user may use any form which conforms to the numeric parameter value described above. Any numeric value format which does not contain a decimal point indicates that the value will be rounded to the nearest integer.

Examples of the transformation of user specified values:

User specified value	Value format	Value used in assignment
123.45	x.xx	123.45
123.45678	x.xx	123.4567
123.45678	xxxx	123
+12	x.xx	12.0
-12	x.xx	-12.0
00000008	x.xx	0.0 (8 is eighth digit)
00000008	xxxx	0
00.000008	x.xx	0.0
123456789	x.xx	1234567.0 (excess digits ignored)
12345678	xxxx	1234567 (excess digits ignored)

In the following parameter value descriptions, character values are shown with the characters exactly as they must be specified. For example PERS indicates that the 4 characters PERS should be entered as the value.

4.2.3 $ASSIGN parameter descriptions

These descriptions show the parameter keyword to the left of the equals sign (=). The values allowed for the parameter are given in a list to the right of the equals sign. The text gives the parameter name (in words) and describes the significance of each item in the list of values.

AOCC=x.xx — Auto occupancy (persons per auto)

=PERS — Default value: 1.00

x.xx specifies the auto-occupancy for trips between all origins and all destinations (global value)

PERS indicates use of the data bank matrix (AUTO OCCUPANCY) specifying the auto-occupancy by O-D pair.

BCAP=xxx. — Bus capacity (persons)

Default value: 60.

xxx. specifies the design capacity of a bus

BEQU=x.xx — Auto equivalent of one bus (automobiles)

Default value: 3.00

x.xx specifies the number of automobiles equivalent to 1 bus for calculating the total traffic volume (automobiles)

BUSF=NO — Bus frequency on road links

=COMP — Default value: NO (except COMP for bimodal assignment)

=OBSx — NO indicates that bus frequency information should not be considered.

COMP indicates that the bus frequency is to be computed from the transit line specifications.

OBSx indicates use of the observed value of the bus frequency on each road link, as found in the user data item x in the road link specifications (buses per hour).

CONT=YES — Continue road traffic assignment (post optimization)

=NO — Default value: NO

This parameter not implemented

DFNC=LOGI — Demand function

=LINE — Default value: LOGI

LOGI indicates use of a logit demand function

LINE indicates use of a linear demand function

DSPD=xx.x — Default transit speed (mph)

Default value: 10.0

xx.x specifies the default transit speed used to calculate the transit travel time, if the time is not specified otherwise and the length of the link can be determined

DTIM=xx.x Default transit travel time (minutes)

Default value: 15.0

xx.x specifies, for transit-only links, the transit travel time used when the travel time cannot be determined from data bank information.

EXPF=x.xx Express bus time factor

Default value: 0.75

x.xx specifies,for transit-only links,the factor used to multiply the normal transit travel time to obtain the travel time for express buses.

HDWY=xxx.,xxx., Permissible transit headways for capacity adjustment (minutes)

Default values: 2., 5., 10., 15., 20., 30.

xxx.,xxx.,... specifies the list of permissible transit headways in ascending order. The headways are those to be used when adjusting transit capacity (see parameter ITCP). The list may contain from 1 to 20 values. The values must be contained on one subcommand card. The decimal point may be omitted for integer values, for example 2, 5, 7.5, 10, 15.

ITCK=xxxx Bimodal assignment, iterations between check points

Default value: undefined

This parameter not implemented.

ITCP=xxxx Iterations, transit capacity

Default value: 0

xxxx specifies the maximum number of bimodal or transit assignment iterations following an automatic modification of transit capacity (headway). If xxxx is zero, the transit capacity modification is not attempted.

This parameter has no effect on road traffic assignments.

ITLT=xxxx Iterations, link times

Default value: 0 (all assignments except bimodal)
: 1 (bimodal assignment)

xxxx specifies the maximum number of bimodal or road traffic with variable demand assignment iterations, following an automatic modification of transit link times. If xxxx is zero or if parameter TIMP=PRED, the transit link time modification is not attempted.

This parameter has no effect on transit-only nor on road traffic with fixed demand assignments.

ITRD=xxxx Iterations within road traffic assignment

Default value: 15

xxxx specifies the maximum iterations within the road traffic assignment procedure. The procedure terminates normally when the maximum number of iterations has been executed.

LTIM=OBS
=PRED

Road link times used to calculate initial transit link times

Default value: OBS

OBS specifies use of the observed road link travel times, if any found in the road link specifications.

PRED specifies use of the road link times predicted by a road traffic assignment, found on file RLTIVL

ODAU=NEW
=OLD

Automobile demand matrix

Default value: OLD

This parameter is only applicable when CONT = YES.

NEW specifies that the O-D matrix of auto trips is different from the one used to predict the travel times and volumes (file RLTIVL) which will be used to restart the assignment.

OLD specifies that the O-D matrix of auto trips is the same as the one used to predict the travel times and volumes (file RLTIVL) which will be used to restart the assignment.

This parameter not implemented.

ODMA=x
=name

O-D matrix number or name

Default value: 1

x specifies the O-D matrix to be used for the assignment. x is the sequence number (1, 2 or 3) of the matrix in the data bank.

name specifies the O-D matrix name (4 characters) entered on the O-D MATRIX data bank header card.

PMUL=YES
=NO

Use transit path multipliers

Default value: NO

YES specifies that transit path multipliers are to be used to weight the wait and access times and in-vehicle times when computing the shortest path by transit.
NO specifies that transit path multipliers are not to be used.

RTIM=xxx

Time limit, road traffic assignment

Default value: 0

The user must set this parameter for any bimodal or road traffic assignment.

xxxx specifies the maximum number of time units (seconds on a CDC system) for the road traffic assignment procedure. The prodecure terminates normally at the end of the iteration during which the time limit is exceeded.

SPLT=x.xx
=AUTO
=PCNT
=PERS
=PRED

Modal split specification

Default value: 1.00 all assignments except transit, fixed demand
: 0.00 transit-only assignment, fixed demand

x.xx specifies a global value for the percentage of total trips made by auto.

AUTO specifies use of the MODAL SPLIT matrix from the data bank. The matrix specifies the number of auto-trips between each O-D pair (AUTO is not accepted for transit only assignments).

PCNT specifies use of the MODAL SPLIT matrix from the data bank. The matrix specifies the percentage of total trips made by auto for each O-D pair.
PERS specifies use of the MODAL SPLIT matrix from the data bank. The matrix specifies the number of person-trips by auto between each O-D pair.
PRED specifies use of a matrix predicted by a previous road traffic assignment (file GPQAU). The matrix specifies the number of person trips by auto between each O-D pair.

TDIV=YES
=NO

Transit line diversion algorithm

Default value: NO

YES specifies that any transit assignment uses the line diversion algorithm.
NO specifies that the line diversion algorithm is not used.

TIMP=COMP
=PRED

Transit impedance specification

Default value: COMP

This parameter is only valid for a road traffic assignment with variable demand.
COMP specifies that the transit impedances are to be calculated from the transit line specifications.
PRED specifies that the transit impedances are to be read from an existing file TZIMP. This file may have been created by a preceding bimodal assignment or a road traffic assignment with variable demand for which TIMP=COMP was specified.

TOLT=.xxx Significant relative difference when comparing link times (tolerance)

Default value: .05 (5 per cent)

.xxx specifies the minimum relative difference considered significant when verifying transit link times changes implied by the preceding road assignment.

This parameter applies only to bimodal or road traffic with variable demand assignments and when parameter ITLT ≠ 0.

TOLU=.xxx Significant relative difference when comparing impedances (tolerance)

Default value: .05 (5 per cent)

.xxx specifies the minimum relative difference considered significant when verifying if transit impedances have changed either due to changes in link time implied by a preceding road traffic assignment (only if parameters ITLT ≠ 0) or due to a change in transit capacity (only if parameter ITCP ≠ 0).

TOLV=.xxx Significant relative difference when comparing auto time and impedance (tolerance)

Default value: .05 (5 per cent)

.xxx specifies the minimum relative difference considered significant when comparing the automobile travel time and impedance (inverse of demand function) in a variable demand road traffic assignment.

This parameter only applies to bimodal or road traffic with variable demand assignments.

TVAL=YES
=NO — Validate transit network on input

Default value: YES

YES specifies that the transit network is to be validated as it is read from the data bank.

NO specifies no such validation.

VADL=xxxx — Variable demand limit (persons)

Default value: 50

In a variable demand or bimodal assignment, any O-D pair having a total demand of less than xxxx persons, will be treated as if its demand were fixed at the level specified by the modal split specification (see SPLT). If xxxx is less than 2, VADL is set to 2. VADL is ignored in transit assignments.

VEAD=YES
=NO — Vehicle adjustment matrix

Default value: NO

YES specifies that the vehicle adjustment matrix is to be included in any road traffic assignment.

NO specifies the matrix is not to be included.

WAIT=x.xx — Transit wait time coefficient

Default value: 1.0

x.xx specifies the coefficient applied to the transit wait time when determining the shortest path by transit.

WALK=x.xx Transit walk time coefficient

Default value: 1.0

x.xx specifies the coefficient applied to the transit walk time when determining the shortest path by transit.

WFAC=.xxx Transit regularity factor

Default value: .5

.xxx specifies the factor applied to the transit headways to determine the transit waiting time.

WMAX=xxx. Maximum transit waiting time (minutes)

Default value: 120.

xxx. specifies the maximum transit waiting time allowed. Any waiting time calculated from the headway, with the regularity factor applied, which exceeds xxx. will be reduced to xxx.

WMIN=xxx. Minimum transit waiting time (minutes)

Default value: 0.

xxx. specifies the minimum transit waiting time allowed. Any waiting time calculated from the headway with the regularity factor applied, which is less than xxx., will be increased to xxx.

WPEN=xx.x Transit waiting penalty (minutes)

Default value: 0.0

xx.x specifies the number of minutes added to the transit waiting time calculated from the headway with the regularity factor applied. xx.x is added to the transit waiting time even if it has been reduced to the value specified by parameter WMAX.

4.3 Sample assignment requests

4.3.1 Bimodal assignment

```
$ASSIGN, BIMODAL
IDENT SAMPLE BIMODAL ASSIGNMENT, NETWORK 1
PARAMETERS TVAL=NO  LTIM=OBS
PARAM AOCC=1.3  ODMA=2  SPLT=.7  RTIM=300
PARAM ITCP=1
```

This assignment request specifies the following:

- The transit network is not validated on input (TVAL=NO).
- Initial transit travel times are calculated based on the observed road link travel times (LTIM=OBS).
- The assignment is performed using OD matrix number 2 from the data bank (ODMA=2) with auto occupancy 1.3 persons per auto between all OD pairs (AOCC=1.3) and with an initial estimate of the modal split of .7 i.e. 70 per cent of the total trips between all OD pairs are made by auto (SPLT=.7).
- Each execution of the road traffic assignment with the bimodal assignment is limited to 300 CPU seconds (RTIM=300).
- Transit capacity is verified once after road traffic and transit assignments (ITCP=1). If the capacity is incorrect, headways are adjusted and the road and transit assignments reexecuted.
- By default (ITLT=1), after the road assignment, transit travel times are checked for significant changes (TOLT) since

they depend on the predicted road link travel times. If the changes are significant the transit travel times and impedances are updated and the road traffic assignment reexecuted. Since ITLT=1 (default) the check is made only after the first traffic assignment.

Note that RTIM must be specified for any bimodal assignment.

4.3.2 Bimodal assignment, minimum specification

```
$AS,BIM
PAR RTIM=300
```

or in non abbreviated form

```
$ASSIGN,BIMODAL
PARAMETERS RTIM=300
```

This assignment request specifies the following:

- Use of the standard computer file identification (no IDENT subcommand)
- Validation of the transit network (default TVAL=YES)
- Observed road link times may be available for some or all links (default LTIM=OBS). Initial transit travel times will be calculated using transit speeds for links which have no observed road link time.
- The assignment is performed using OD matrix number 1 (default ODMA=1) with auto occupancy 1.0 (default AOCC=1.0) and with an initial modal split estimate of 1.0 (default SPLT=1.0) i.e. all trips made by auto.
- Each execution of the road traffic assignment is limited to 300 secs (RTIM=300) or to 15 iterations (default ITRD=15) whichever is reached first.
- transit capacity is not verified or corrected after road traffic and transit assignments (default ITCP=0).
- after the first road traffic assignment, transit travel times are checked for significant changes as described in 4.3.1 (default ITLT=1).

4.3.3 Road traffic assignment with fixed demand

```
$ASSIGN,ROAD,FIXED
IDENT SAMPLE FIXED DEMAND TRAFFIC
PARAMETERS ODMA=3   RTIM=500
```

This assignment request specifies the following

- The assignment is performed using OD matrix 3 from the data bank (ODMA=3) with auto occupancy 1.0 (default AOCC=1.0) and modal split estimate of 1.0 (default SPLT=1.0) i.e. OD matrix 3 specifies the number of vehicles to be assigned to the road network.
- The road traffic assignment is limited to 300 CPU secs or to 15 iterations (default ITLT=15) whichever is reached first.

Note that RTIM must be specified for any road traffic assignment.

4.3.4 Road traffic assignment with variable demand

```
$ASSIGN,ROAD,VARIABLE
IDENT SAMPLE ASSIGNMENT RUN 53
PARAMETERS ODMA=1  AOCC=PERS  SPLT=PCNT
PARAM RTIM=600  ITRD=20
```

The assignment request specifies the following:

- The assignment is performed using the total OD matrix 1 from the data bank with an auto occupancy matrix from the data bank used to give the number of persons per auto for each OD pair (AOCC=PERS). The initial modal split estimate is given by the modal split matrix in the data bank and it specifies the percentage (fraction) of the total trips made by auto between each OD pair.
- The assignment will apply a logit demand function (default DFNC=LOGI) to OD pairs having at least 50 trips (default VADL=50) in OD matrix number 1. For OD pairs with less than 50 trips, the number of trips assigned will always be the number calculated as the initial modal split estimate (the demand function is not applied).
- The road traffic assignment is limited to 600 CPU seconds (RTIM=600) or to 20 iterations (ITRD=20) whichever is reached first.

Note that RTIM must be specified for any road traffic assignment.

4.3.5 Transit assignment with fixed demand

```
$ASS,TRANSIT,FIXED
IDENT FIXED DEMAND TRANSIT SPLIT .8
PARAM LTIM=PRED  SPLT=0.8
PARAM WPEN=5.0  BCAP=75
```

This assignment request specifies the following:

- Predicted car times will be read from file RLTIVL in order to calculate transit link times (LTIM=PRED) and user-supplied transit-only link times will be multiplied by 0.75 for express lines, if applicable (default EXPF=0.75).
- 20% of the trips (SPLT=0.8) contained in OD matrix 1 (default ODMA=1) will be assigned on the transit network
- A fixed penalty of 5 minutes will be added to all waiting times (WPEN=5.0).
- A transit vehicle design capacity of 75 passengers will be used to calculate bus line statistics. Note that headways will not be adjusted (default ITCP=0).

4.3.6 Transit assignment with variable demand

```
$ASSIGN,TRANSIT,VARIABLE
IDENT VARIABLE TRANSIT SCENARIO 2
PARAM TVAL=NO  EXPF=1.0
PARAM HDWY=5,15,30  BCAP=80  ITCP=1
```

This assignment request specifies the following:

- The transit network will not be validated on input.
- The user-supplied transit-only link times already take into account express service (EXPF=1.0).
- The assignment will apply a logit demand function (default DFNC=LOGI) to OD pairs having at least 50 trips (default VADL=50) in OD matrix number 1 (default ODMA=1). OD pairs with less than 50 trips will not be assigned on the transit network (since default SPLT=1.0).
- After the first assignment, headways will be adjusted using a bus capacity of 80 passengers (BCAP=80) and three permitted headways:5,15 and 30 minutes.
- Then another transit assignment will take place using the new headways (ITCP=1).

5. $COMMENT Print a comment line

Command format:
$COMMENT <any text>

$COMMENT is provided to allow comments in the command sequence. The entire command is reproduced on the output listing when the commands are processed. It has no other effect on the execution of EMME.

Sample commands:

```
$COMMENT  ANY TEXT
$CO   THIS SHOWS  SHORTENED  FORM OF COMMAND CODE
```

Note that $COMMENT is a command. This implies that it terminates the sequence of preceding subcommands.

6. $DATABANK Perform databank operations

Command format:
$DATABANK, <option>

$DATABANK performs databank operations. The operations are requested by subcommands some of which may be followed by headers and data to be processed.

$DATABANK and its subcommand codes may only be abbreviated to their first four or eight characters. For example the following would be:

Valid	Not valid
$DATABANK	$DA
$DAT	$DATA
$DATABAN	
CORRECTION	COR
CORR	CORRECT
CORRECTI	

The same rules apply to any word of a header record (see 6.2.1.2).

6.1. $DATABANK option

The only option recognized is NEW.

NEW indicates that the subsequent subcommands, headers and data are to create a new databank.

If NEW is omitted, the subsequent subcommands and any headers and data will be processed using the existing databank.

6.2. $DATABANK subcommands

The following subcommands are recognized:

CORRECTION
SCENARIO
USE
PRINT
PUNCH
PURGE

The subcommands are processed in the order in which they are encountered. They may be given in any order except for the CORRECTION subcommand which may not appear after a USE subcommand.

6.2.1. CORRECTION Request addition, modification or deletion of data

The subcommand has the following form:

CORRECTION {B, } <option>

that is the subcommand code followed by 1 or more blanks or commas which may be followed by an option.

The CORRECTION subcommand is followed by 1 or more sets of data and/or 1 or more INCLUDE records. Each set consists of

- a header record which defines the type of data to be processed
- 1 or more data records in the format implied by the header record
- an END record

The use of an INCLUDE record (see paragraph 6.2.1.3.) following a CORRECTION subcommand causes processing of set(s) of data previously entered as a SCENARIO (see paragraph 6.2.2.).

Figures 6-1 and 6-2 show examples of a CORRECTION subcommand and the headers, data and INCLUDE records.

The operation to be performed (Addition, Modification or Deletion) is specified as A, M or D in the first column of the data records and applies to all the non-blank data in the record. If the first column is blank, it is interpreted as an A. Blank data fields are ignored.

For Addition, the non-blank data in the data record must not already exist in the databank, otherwise an error message will be printed and the data record rejected.

For Modification, the non-blank data in the data record is used to modify the entity indicated in the record (road link, transit line, etc.). Individual data items within a record may be deleted by filling the field for that item with minus signs (-).

For Deletion, all the data previously entered into the databank for the entity indicated in the data record (road link, transit line, line of an O-D matrix), is deleted.

For further information on these operations see the EMME User Guide.

The CORRECTION subcommand causes the databank to be modified according to the operations (A, M or D) and the data in each record of each set of data (unless an error is detected in the record). These modifications are permanent.

```
$DATABANK
CORRECTION
TRANSIT ACCESS LINKS
A  1  202*   20
M  2  11     17
D  3  12
D 12   2
END
NODE EXPANSION NOLIST
  524  37  38  39  40
  525  18
END
```

Figure 6-1: Use of CORRECTION subcommand

```
$DATABANK
CORRECTION
INCLUDE NEW-HWY
ROAD LINKS
D  1  11*
M 55  28  08
END
```

Figure 6-2: Use of INCLUDE within a CORRECTION

6.2.1.1. CORRECTION options

The following options are recognized:

LIST
NOLIST

If the option is omitted, LIST is assumed.

LIST indicates that the data records within each following set of data are to be printed as they are read in.

NOLIST indicates that the data records are not to be printed. If a data record contains an error, the record will be printed together with the message indicating the error.

Note that LIST may be overridden by a smiliar option in a header (see paragraph 6.2.1.2.) but NOLIST may not.

Header records are always printed as they are read in.

6.2.1.2. Headers and data records

A header record is the first record of a set of data which may follow a CORRECTION (or SCENARIO, see paragraph 6.2.2.) subcommand.

A header specifies a section (also called record) of the databank.

A header record has the following form

<header> $\{^{b}_{,}\}$ <option>

that is a header followed by 1 or more blanks or commas which may be followed by an option.

The only option recognized is NOLIST which indicates that the data records following this header are not to be printed until an END record is reached (unless an error is detected).

The only use of the NOLIST option is to override the LIST option of the preceding subcommand.

The first header record under a CORRECTION subcommand causes the deactivation of all the scenarios that are currently active due to a previous USE subcommand (see paragraph 6.2.3.)

The format of the data records which follow each header depends on the header. These formats are shown on coding forms contained in Appendix A.

Figure 6-3 contains a list of the headers currently recognized together with the number and name of the coding form showing the format of the data records which must follow the header.

Any word in a header may be abbreviated to its first 4 or 8 characters. For example:

SOCIONOMIC COEFFICIENTS FORMAT 1

can be abbreviated to:

SOCIONOM COEF FORM 1

Information on filling out the forms can be found in the EMME User Guide.

HEADER	CODING FORM	
	no	name
AUTO OCCUPANCY	03	od matrix
CENTROIDS	01	Centroid
DEMAND FUNCTION COEFFICIENTS	70	demand function travel time coefficient
LINES	50	transit line
MODAL SPLIT	03	od matrix
NODE COORDINATES	02	node coordinate
NODE EXPANSION	51	transit node - road nodes correspondence
O-D MATRIX n aaaa	03	od matrix
ROAD LINKS	60	road link
SCALE x.xx	--	
SOCIONOMIC COEFFICIENTS FORMAT 1	72	socio-economic variable/coeff. form 1
SOCIONOMIC COEFFICIENTS FORMAT 2	71	socio-economic variable/coeff. form 2
SOCIONOMIC VARIABLES FORMAT 1	72	socio-economic variable/coeff. form 1
SOCIONOMIC VARIABLES FORMAT 2	71	socio-economic variable/coeff. form 2
TRANSIT ACCESS LINKS	53	transit access link
TRANSIT LINKS	52	transit link
TRANSIT PATH MULTIPLIERS	54	transit path multiplier
VEHICLE ADJUSTMENT MATRIX	03	od matrix
ZONE GROUPS	04	zone group

Figure 6-3: List of $DATABANK headers and corresponding coding form number and name

6.2.1.3. INCLUDE records

The record has the following form:

INCLUDE name

'name' is the name of a scenario which was previously entered into the databank (see SCENARIO subcommand, paragraph 6.2.2.)

The INCLUDE record causes the headers and data from the scenario 'name' to be processed when the INCLUDE record is encountered.

If the INCLUDE record follows a CORRECTION subcommand, the headers and data from the scenario are processed to permanently modify the databank. The scenario itself remains unchanged.

If the INCLUDE record follows a SCENARIO subcommand, the INCLUDE record is stored in the scenario as scenario data.When the scenario is used via a USE subcommand, the header and data from the INCLUDEd scenario are processed when the INCLUDE record is encountered.

6.2.1.4. Header SCALE

The SCALE header record is not followed by data records nor by an END record. The format is as follows

SCALE $\{\overset{\text{b}}{,}\}$ s

that is the header followed by one or more blanks or commas, followed by a real number, 's', which defines the scale.

If a scale had previously been defined, it is replaced by 's'. If 's' consists of one or more minus signs (-) only, the current scale is deleted.

6.2.1.5. Header 0-D MATRIX

This header record has the following form:

$$\text{0-D } \textvisiblespace \text{ MATRIX } \textvisiblespace \text{ n } \{^{\textvisiblespace}_{,}\} \text{ name}$$

Note that 0-D may not contain blanks.

'n' may be 1, 2 or 3 and may not be omitted. 'name' is a name associated with matrix number 'n' which consists of 1 to 4 characters. 'name' may be omitted. Once a name has been defined for matrix 'n', the same name (or blank) must be used on subsequent headers which also refer to matrix 'n'.

6.2.1.6. END record

The END record has the following form

END

the END record indicates the end of the data records for the current header. It may not be omitted except for SCALE (see paragraph 6.2.1.4.)

6.2.2. SCENARIO Enter scenario data into the databank

The subcommand has the following form

SCENARIO {b} name {b} <option>

that is the subcommand code followed by 1 or more blanks or commas followed by the 'name' of the scenario which may be followed by 1 or more blanks or commas and an option.

The name of the scenario can be from 1 to 8 characters long and identifies the scenario data in the databank.

The SCENARIO subcommand is followed by 1 or more sets of data and / or 1 or more INCLUDE records as described in paragraph 6.2.1., 6.2.1.2 and 6.2.1.3. Moreover the sets of data are not used to modify the data bank immediately but are stored in the databank as a 'scenario' with the 'name' specified. The scenario can be used via a USE subcommand (see paragraph 6.2.3.) or an INCLUDE subcommand (see paragraph 6.2.1.3.).
The headers and data records which follow a SCENARIO subcommand are checked for syntax errors as they are read and messages are printed for any errors. Any records which contain errors are not stored in the scenario data.

Figure 6-4 shows an example of a SCENARIO subcommand, header and data.

6.2.2.1. SCENARIO options

The following options are recognized

LIST
NOLIST

If the option is omitted, LIST is assumed. See paragraph 6.2.1.1. for an explanation of the options.

```
$DATABANK
SCENARIO NEW-HWY
ROAD LINKS
D  54   53*
D  53   52*
D  52   47*
D  46   47*
A  54   52* 16   2.5
A  52   46* 16   4.7
END
```

Figure 6-4: Example of a SCENARIO subcommand

6.2.3. USE Use a scenario

The subcommand has the following form

USE {b} name {b} <option>

that is the subcommand code followed by 1 or more blanks or commas and the 'name' of a scenario which may be followed by 1 or more blanks or commas and an option.

The USE subcommand causes the databank to be temporarily modified according to the headers and the operations and data contained in the scenario data identified by the scenario 'name' as described in paragraph 6.2.1. The scenario (data) was previously entered as the result of processing a SCENARIO subcommand and subsequent headers and data (see paragraph 6.2.2.). Note that a scenario may include INCLUDE subcommands in order to apply the data modifications stored as a separate scenario(s).

The modifications to the databank remain in effect until either a $END command is encountered unless this command contains the KEEP option (see paragraph 8.1.) or a CORRECTION subcommand is encountered.

USE subcommands are processed in the order they are encountered and the temporary databank modifications resulting from one USE subcommand remain in effect when another USE subcommand is processed.

6.2.3.1. USE options

The following options are recognized

LIST
NOLIST

If the option is omitted, LIST is assumed. See paragraph 6.2.1.1. for an explanation of the options.

6.2.4. PRINT Print data from the databank

The subcommand has 3 forms

PRINT header
PRINT SCENARIO name
PRINT SCENARIO LIST

6.2.4.1. PRINT header Print data of specified type

'header' is any one of the headers listed in Figure 6-3.

The 'PRINT header' subcommand requests printing of the data indicated by 'header', that is all the data which followed the same header and was entered into the databank as a result of CORRECTION and/or USE subcommands.

For example

PRINT ROAD LINKS

requests printing of the road link data, previously entered following a ROAD LINKS header.

If scenarios have been USEd and are still in effect (see paragraph 6.2.3.), any temporary modifications to the data are shown in the printout and any data record, which has been temporarily deleted, is marked as inactive in the 'status' column.

6.2.4.2. PRINT SCENARIO name Print scenario data

'name' is the name of a scenario which was previously entered into the databank (see SCENARIO subcommand, paragraph 6.2.2.)

The PRINT SCENARIO subcommand requests printing of headers and data previously entered following the 'SCENARIO name' subcommand.

For example

PRINT SCENARIO 1984

requests printing of the header and data in the scenario '1984'.

6.2.4.3. PRINT SCENARIO LIST Print list of scenario names

This subcommand requests printing of a list of the names of all scenarios previously entered into the databank as a result of SCENARIO subcommands (see paragraph 6.2.2.).

6.2.5. PUNCH Punch data from the databank

The subcommand has 2 forms:

PUNCH header
PUNCH SCENARIO name

Note that operating system control cards can be used to assign the file on which the data is punched (PCHFIL) to a tape.

6.2.5.1. PUNCH header Punch data of a specified type

'header' is any one of the headers listed in Figure 6-3.

The 'PUNCH header' subcommand requests the punching of the section of the databank corresponding to the given 'header'. The result is a set of data (in terms of paragraph 6.2.1.) which could be used under a CORRECTION subcommand to define the same data section in a new databank.
The set of data punched consists of a header record followed by data records and terminated by an END record. The data records are punched in the format defined by the corresponding coding form as indicated in Figure 6-3. The operation code (first column, see 6.2.1.) is left blank.

For example

PUNCH ROAD LINKS

requests the punching of all the currently active road link data in the format of coding form number 60.

If scenarios have been USEd and are still in effect (see paragraph 6.2.3.), any temporary modifications to the data will be included in the punched version but data records which have been temporarily deleted are not punched.

6.2.5.2. PUNCH SCENARIO name Punch scenario data

'name' is the name of a scenario which was previously entered into the databank (see SCENARIO subcommand, paragraph 6.2.2.)

The PUNCH SCENARIO subcommand request punching of all headers and data previously entered following the 'SCENARIO name' subcommand.

For example

PUNCH SCENARIO HWY2

requests punch of the headers and data in the scenario 'HWY2'.

6.2.6. PURGE Delete data from data bank

The subcommand has 2 forms:

PURGE header
PURGE SCENARIO name

6.2.6.1. PURGE header Delete all data of a specified section

'header' is any one of the headers listed in Figure 6-3.

The 'PURGE header' subcommand requests deletion of all the data indicated by 'header', that is all data which followed the same header and was entered into the databank as a result of CORRECTION and/or USE subcommands

For example

PURGE DEMAND FUNCTION COEFFICIENTS

requests that all the demand function coefficients be deleted from the databank.

6.2.6.2. PURGE SCENARIO name Delete scenario data

'name' is the name of a scenario which was previously entered into the databank (see SCENARIO subcommand, paragraph 6.2.2.).

The PURGE SCENARIO subcommand requests that the scenario, including all the headers and data previously entered following the SCENARIO subcommand, be deleted from the databank.

For example

PURGE SCENARIO VITAL

requests deletion of the scenario 'VITAL'

Note that the scenario name is removed from the list of scenarios.

7. $DEBUG Control debugging output

Command format:
$DEBUG {,} <option>

$DEBUG controls the printing of information (debugging output) which is intended as an aid in debugging EMME. If no $DEBUG command has been encountered debugging output is suppressed. $DEBUG may be entered several times in a command sequence to change the option desired for the execution of different commands.

7.1 $DEBUG options

The options recognized are:

option	effect
YES	Debugging output is printed
NO	Debugging output is suppressed

Sample commands:

```
$DEBUG,YES
$DE  NO
```

8. $END Terminate EMME

Command format:
$END {,} <option>

$END indicates the end of a sequence of commands. The system EMME is terminated.

8.1. $END option

The only option recognized is KEEP.

KEEP indicates that all active scenarios will be kept active for the next use of the databank.

If KEEP is omitted, all active scenarios (if any) are deactivated before the databank is closed.

9. $REPORT Print reports

Command format:
$REPORT, <option>, <option>

$REPORT requests the printing of reports. Since the files used to produce the reports may depend on the type of assignment which produced them, $REPORT recognizes the same options and subcommands as $ASSIGN. In addition subcommands unique to $REPORT request the reports required.

Note that few of the options and parameters specified on the PARAMETERS subcommand are used by the report programs but they are all accepted so that the user may use the options and subcommands from the $ASSIGN command. If the $REPORT command is preceded in the same execution of EMME, by any $ASSIGN or $REPORT command, the options and parameters set when those commands were executed remain in effect, otherwise the default values are assumed for the options and parameters not specified.

9.1. $REPORT options

The options recognized are:

Option	Type of assignment
BIMODAL	Bimodal
ROAD	Road traffic only
TRANSIT	Transit only
FIXED	Fixed demand
VARIABLE	Variable demand

Valid combinations of these options i.e. assignments which produced the files used for the reports are as described in paragraph 4.1. If no option

is specified, the option(s) specified or a preceding $ASSING or $REPORT are used and if there was no preceding $ASSIGN nor $REPORT the option BIMODAL is assumed.

9.2. $REPORT subcommands

The following subcommands are recognized:

IDENT
PARAMETERS
LINK
DEMAND
AUTO TIME DISTRIBUTION

9.2.1. IDENT Set computer file identification

IDENT is accepted only for compatibility with $ASSIGN subcommands. It has no effect on the reports.

9.2.2. PARAMETERS Set assignment parameters

The PARAMETERS subcommand is as described for the $ASSIGN command in paragraph 4.2.2. However, if this $REPORT was preceded by a $ASSIGN or a $REPORT the parameter values from these commands and any PARAMETERS subcommands remain in effect and only the values specified on the current PARAMETERS are changed. If there was no preceding $ASSIGN or $REPORT the parameter values are set to the default values as described in paragraph 4.2.3.

EMME EXAMPLE
BIMODAL ASSIGNMENT DEFAULT TRANSIT SPEED = 100.

ROAD TRAFFIC ASSIGNMENT

LINK	FROM	TO	LENGTH (MILES)	TYPE	FREE FLOW (MPH)	CAPACITY (VEH/HR)	OBSERVED TIME (MINS)	OBSERVED VOLUME (VEH/HR)	PREDICTED SPEED (MPH)	TRAVEL TIME (MINS)	AUTO VOLUME (VEH/HR)	LANES	BUS VOLUME (VEH/HR)
1	1	31	2.00	1	10.00	700.			10.00	12.00	635.0	1.0	0.0
2	1	55	1.05	1	10.00	700.			10.00	6.30	441.0	1.0	0.0
3	2	11	1.60	1	10.00	700.			10.00	9.60	340.0	1.0	0.0
4	2	12	1.80	1	10.00	700.			10.00	10.80	823.0	1.0	0.0
5	2	56	1.20	1	10.00	700.			10.00	7.20	778.1	1.0	0.0
6	4	13	2.45	1	10.00	700.			10.00	14.70	0.0	1.0	0.0
7	4	14	1.30	1	10.00	700.			10.00	7.80	223.0	1.0	0.0
8	4	19	2.75	1	10.00	700.			10.00	16.50	880.0	1.0	0.0
9	5	12	.65	1	10.00	700.			10.00	3.90	574.0	1.0	0.0
10	9	8	.80	1	10.00	700.			10.00	4.80	693.1	1.0	0.0
11	9	24	.80	1	10.00	700.			10.00	4.80	786.1	1.0	0.0
12	9	51	.70	1	10.00	700.			10.00	4.20	0.0	1.0	0.0
13	9	52	1.15	1	10.00	700.			10.00	6.90	93.0	1.0	0.0
14	9	53	1.70	1	10.00	700.			10.00	10.20	0.0	1.0	0.0
15	9	48	1.85	1	10.00	700.			10.00	11.10	0.0	1.0	0.0
16	11	12	1.60	1	10.00	700.			10.00	9.60	0.0	1.0	0.0
17	12	55	1.05	2	17.50	500.			16.85	3.74	340.0	2.0	0.0
18	12	13	1.80	1	10.00	700.			10.00	10.80	0.0	1.0	0.0
19	12	24	1.60	2	17.50	500.			8.13	19.19	823.0	2.0	0.0
20	12	8	2.45	1	10.00	700.			10.00	14.70	0.0	1.0	0.0
21	12	56	1.15	2	17.50	500.			17.50	3.94	0.0	2.0	0.0
22	13	8	1.30	1	10.00	700.			10.00	7.80	0.0	1.0	0.0
23	13	16	1.60	2	17.50	500.			17.40	5.52	223.0	2.0	0.0
24	15	16	1.25	2	17.50	500.			17.50	4.29	0.0	2.0	0.0
25	15	14	1.60	2	17.50	500.			17.50	5.49	0.0	2.0	0.0
26	16	15	1.25	2	17.50	500.			17.50	4.29	0.0	2.0	0.0
27	16	17	1.50	2	17.50	500.			17.50	5.15	0.0	2.0	0.0
28	16	18	1.85	2	17.50	500.			17.38	6.39	223.0	2.0	0.0
29	17	16	1.50	2	17.50	500.			17.50	5.15	0.0	2.0	0.0
30	17	18	1.05	2	17.50	500.			17.50	3.60	0.0	2.0	10.0
31	17	43	2.40	2	17.50	500.			17.50	8.23	0.0	2.0	0.0
32	18	19	1.85	2	17.50	500.			17.50	6.34	0.0	2.0	0.0
33	18	17	1.05	2	17.50	500.			17.50	3.60	0.0	2.0	0.0
34	18	19	1.50	2	17.50	500.			17.50	5.14	0.0	2.0	10.0
35	18	42	1.70	2	17.50	500.			15.98	6.40	44.0	2.0	0.0
36	19	6	2.75	1	10.00	700.			10.00	16.50	0.0	1.0	0.0
37	19	18	1.50	2	17.50	500.			17.38	5.18	219.0	2.0	0.0
38	19	20	.70	2	17.50	500.			12.11	3.47	0.0	2.0	10.0
39	19	41	1.40	2	17.50	250.			11.33	7.42	165.0	1.0	0.0
40	20	19	.70	2	17.50	500.			16.73	2.51	0.0	2.0	0.0
41	20	21	.80	2	17.50	750.			10.45	2.92	575.0	3.0	10.0
42	20	54	1.40	2	17.50	500.			12.20	6.89	643.0	2.0	0.0
43	20	56	1.80	2	17.50	500.			17.50	6.17	0.0	2.0	0.0
44	21	20	.80	2	17.50	750.			17.49	2.74	175.1	3.0	0.0
45	21	22	.75	2	17.50	750.			17.40	2.59	292.0	3.0	13.0

Figure 9-1

9.2.3. LINK Print road link report

This subcommand has the following form

Link {b / ,} <keyword = value> b <keyword = value>

that is the subcommand code, one or more blanks or commas, followed by 'keyword = value' pairs separated by one or more blanks.
A keyword is a 4 character identifier for a parameter which the user may specify as having the value which follows the equals sign (required). For example:

CORR=LINK

The form of values is as described in paragraph 4.2.2.

The LINK subcommand requests printing of the road link report. This report contains 1 line for each road link as shown in Figure 9-1. The links are sorted in ascending sequence by their origin and destination nodes. The report may contain all the links in the road network or selected links depending on the parameter CORR.

9.2.3.1. LINK parameter description

LINK currently accepts 1 parameter CORR (corridor specification) which may be omitted or set to LINK or NODE.

If CORR= is omitted the link report will contain all links on the road network.

If CORR=LINK the LINK subcommand is followed by 1 or more sets of corridor specifications (see below) which consist of a list of the links to be selected for the report. One report is printed for each set of corridor specifications.

If CORR=NODE the LINK command is followed by 1 or more sets of corridor specifications each of which consists of a list of nodes. The links selected are those which have both nodes in the node list. One report is printed for each set of corridor specifications.

The corridor specifications which may follow the LINK subcommand consists of sets. Each set contains a corridor name record followed by the link or node list (1 or more records) followed by an END record. The format of the records is as follows.

Corridor name record:

Columns	Format	Contents
1-8	2A4	'CORRIDOR'
9	1X	blank
10-13	A4	corridor name (4 characters, printed on link report)

Link or node list records:

Columns	Format	Contents
1	1X	blank
2-5	A4	corridor identifier (not checked or used in report)
6	1X	blank
7-10	I4	Node number 1
12-15	I4	Node number 2
17-20	I4	Node number 3
22-25	I4	Node number 4

Link or node list records: (continued)

Columns	Format	Contents
27-30	I4	Node number 5
.		
.		
.		
57-60	I4	Node number 11
62-65	I4	Node number 12

For a link list the node numbers are taken in pairs. Node numbers 1 and 2 specify a link, 3 and 4 another link, etc...

Less than 12 node numbers may be entered on a record. The first blank (or zero) node number encountered when scanning from left to right terminates processing of the record. Node numbers following a blank or zero node number field are not processed. Subsequent link or node list records are processed however.

END record:

Columns	Format	Contents
1-4	A4	'END '

9.2.4. DEMAND Print predicted auto demand report

This subcommand has the following form:

DEMAND {␣,}

that is a subcommand code followed by one or more blanks or commas. The DEMAND subcommand requests printing of the report on predicted demand by automobile between each O-D pair. The report is printed in a double matrix format as shown in Figure 9-2. Each left hand matrix shows the demand by automobile (persons) form one origin to all destinations for which the total demand is non zero. The right hand matrix shows the demand by automobile as a percentage of the total demand (zero indicates zero total demand, -1.0 indicates zero demand by auto but non zero total demand).

EMME EXAMPLE
BIMODAL ASSIGNMENT DEFAULT TRANSIT SPEED = 100.
PREDICTED AUTO DEMAND

OD MATRIX ROW NUMBER 1 NUMBER OF NON ZERO VALUES 4

	PREDICTED AUTO DEMAND (PERSONS)										PREDICTED AUTO DEMAND (PERCENTAGE OF TOTAL)									
CENTROID	00	1	2	3	4	5	6	7	8	9	CENTROID	00	1	2	3	4	5	6	7	8	9
0		0	0	0	0	0	394	175	270	186	0		0.00	0.00	0.00	0.00	0.00	.84	.96	.93	.93

OD MATRIX ROW NUMBER 2 NUMBER OF NON ZERO VALUES 4

	PREDICTED AUTO DEMAND (PERSONS)										PREDICTED AUTO DEMAND (PERCENTAGE OF TOTAL)									
CENTROID	00	1	2	3	4	5	6	7	8	9	CENTROID	00	1	2	3	4	5	6	7	8	9
0		0	0	0	0	0	477	189	298	209	0		0.00	0.00	0.00	0.00	0.00	.99	.99	.99	1.00

OD MATRIX ROW NUMBER 3 NUMBER OF NON ZERO VALUES 4

	PREDICTED AUTO DEMAND (PERSONS)										PREDICTED AUTO DEMAND (PERCENTAGE OF TOTAL)									
CENTROID	00	1	2	3	4	5	6	7	8	9	CENTROID	00	1	2	3	4	5	6	7	8	9
0		0	0	0	0	0	253	190	105	205	0		0.00	0.00	0.00	0.00	0.00	.87	.95	.96	.93

OD MATRIX ROW NUMBER 4 NUMBER OF NON ZERO VALUES 4

	PREDICTED AUTO DEMAND (PERSONS)										PREDICTED AUTO DEMAND (PERCENTAGE OF TOTAL)									
CENTROID	00	1	2	3	4	5	6	7	8	9	CENTROID	00	1	2	3	4	5	6	7	8	9
0		0	0	0	0	0	298	209	318	229	0		0.00	0.00	0.00	0.00	0.00	.94	.99	1.00	.99

OD MATRIX ROW NUMBER 5 NUMBER OF NON ZERO VALUES 4

	PREDICTED AUTO DEMAND (PERSONS)										PREDICTED AUTO DEMAND (PERCENTAGE OF TOTAL)									
CENTROID	00	1	2	3	4	5	6	7	8	9	CENTROID	00	1	2	3	4	5	6	7	8	9
0		0	0	0	0	0	175	21	153	226	0		0.00	0.00	0.00	0.00	0.00	.83	.10	.83	.94

Figure 9-2

9.2.5. AUTO TIME DISTRIBUTION Print automobile trip time distribution

This subcommand has the following form:

AUTO {B,} TIME DISTRIBUTION

that is a subcommand code AUTO followed by one or more blanks or commas. The text TIME DISTRIBUTION is optional and is not checked.

The AUTO subcommand requests printing of the automobile trip time distribution report. This report is based on the trip time between each origin and destination (centroid) by automobile. The trip times are classified in the ranges indicated on the report and the distribution represents the total (predicted) demand for trips in each trip time range.

10. $TITLE Enter title line text

Command format:
$TITLE {b,} n {b,} <text>

$TITLE replaces report title line n with <text>

where n = 1 or 2 specifies titles line 1 or 2

<text> is any text, maximum 60 characters

n and <text > are separated by 1 blank or comma (separator). The character or blank which follows this separator determines how the <text> is inserted into the title line n. If the first character in <text> is non-blank, the text is centered in the report title even if less than 60 characters are given (embedded blanks are retained). If the first character of <text> is blank, the 60 characters (including blanks) starting with this blank are centered in the report title (if there are less than 60 characters from this blank to the end of the record, blanks are added to fill the 60 characters).

The 2 title lines are initially filled with blanks.

$TITLE commands may be entered several times in a command sequence to change the title lines on reports produced by the execution of other commands.

Appendix A

EMME Coding forms

EMME

PROJECT EXAMPLE

CODED BY CLAUDE ACHIM

CENTROID CODING FORM

DATE 9-17-76 PAGE 1 OF 1

	CENTROID	X COORDINATE	Y COORDINATE	NAME OF ZONE	CENTROID	X COORDINATE	Y COORDINATE	NAME OF ZONE	USER INFORMATION
1	1			UN					
2	2			DEUX					
3	3			TROIS					
4	4			QUATRE					
5	5			CINQ					
6	6			SIX					
7	7			SEPT					
8	8			HUIT					
9	9			NEUF					
10									
11									
12									
13									
14									
15									
16									
17									
18									
19									
20									
21									
22									
23									
24									
25									

emme-01-0

27-7-76

EMME

PROJECT EXAMPLE

CODED BY CLAUDE ACHIM

NODE COORDINATE CODING FORM

DATE 9-17-76 PAGE 1 OF 1

	NODE	X COORDINATE	Y COORDINATE	NODE	X COORDINATE	Y COORDINATE	NODE	X COORDINATE	Y COORDINATE	NODE	X COORDINATE	Y COORDINATE	USER INFORMATION
1	31	0	100	32	10	60	33	80	60				
2	34	120	60	35	150	60	36	180	60				
3	41	220	60										
4	42	240	60										
5	43	300	60										
6	52	130	10										

emme-02-0

27-7-76

EMME

PROJECT EXAMPLE

CODED BY CLAUDE ACHIM

OD MATRIX CODING FORM

DATE 9-17-76 PAGE 1 OF 1

	Grp/Mat	ORIGIN CENTROID	DESTN CENTROID	=	TRIPS	DESTN CENTROID	=	TRIPS	DESTN CENTROID	=	TRIPS	DESTN CENTROID	=	TRIPS	DESTN CENTROID	=	TRIPS	DESTN CENTROID	=	TRIPS	USER INFORMATION
	1 2 3 4 5	6	7 8 9 10	11	12 13 14 15 16	17 18 19 20	21	22 23 24 25 26	27 28 29 30	31	32 33 34 35 36	37 38 39 40	41	42 43 44 45 46	47 48 49 50	51	52 53 54 55 56	57 58 59 60	61	62 63 64 65 66	67 68 69 70 71 72	73 74 75 76 77 78 79 80
1		1	6	=	470	7	=	180	8	=	290	9	=	200		=			=		
2		2	6	=	480	7	=	190		=			=			=			=		
3		9	4	=	100		=			=			=			=			=		
4		5	7	=	220	9	=	240		=			=			=			=		
5				=			=			=			=			=			=		
6				=			=			=			=			=			=		
7				=			=			=			=			=			=		
8				=			=			=			=			=			=		
9				=			=			=			=			=			=		
10				=			=			=			=			=			=		
11				=			=			=			=			=			=		
12				=			=			=			=			=			=		
13				=			=			=			=			=			=		
14				=			=			=			=			=			=		
15				=			=			=			=			=			=		
16				=			=			=			=			=			=		
17				=			=			=			=			=			=		
18				=			=			=			=			=			=		
19				=			=			=			=			=			=		
20				=			=			=			=			=			=		
21				=			=			=			=			=			=		
22				=			=			=			=			=			=		
23				=			=			=			=			=			=		
24				=			=			=			=			=			=		
25				=			=			=			=			=			=		

EMME

ZONE GROUP CODING FORM

PROJECT EXAMPLE

CODED BY CLAUDE ACHIM

DATE 9-17-76 PAGE 1 OF 1

	GROUP	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	ZONE	USER INFORMATION
1	E001	1	3	5											
2	E002	2	4	6											
3	E003	7	8												

emme-04-0

24-11-76

EMME

PROJECT ..EXAMPLE.....

CODED BY ..CLAUDE..ACHIM..

TRANSIT LINE CODING FORM

DATE ..9-17-76... PAGE .1.. OF .7..

UPDATE CODE (A,M,D)

LINE NO. 1

NAME BEYOND ROAD

HEADWAY 20.0

LENGTH 19.

SPEED 15.

TYPE

LAYOVER

VEHICLES

USER INFORMATION

NODE SEQUENCE

LINE	NODE	NODE	NODE	NODE	NODE	NODE	NODE	NODE	NODE	NODE	NODE	NODE	NODE	NODE	USER INFORMATION
1	101	11	55	103	110	33	34	35	36A	22R	21				
2															
3															
4															
5															
6															
7															
8															
9															
10															
11															
12															
13															
14															
15															
16															

emme-50-0

27/7/76

EMME

CODED BY ..CLAUDE..ACHIM....

PROJECT ..EXAMPLE.....

TRANSIT NODE - ROAD NODES CORRESPONDANCE

DATE ..9-17-76..... PAGE ..1..OF ..1...

	TRANSIT NODE	CORRESPONDING ROAD NODES	USER INFORMATION
1	101		
2	102		
3	103	0	
4	104	0	
5	105	0	
6	106		
7	110	27 28 29 30	
8	112	23 24 25 26	
9	126	37 38 39 40	
10	521	33	
11	522	35	
12	523	36	
13	524	37 38 39 40	
14	525	18	
15			
16			
17			
18			
19			
20			
21			
22			
23			
24			
25			

emme-51-0

27-7-76

PROJECT EXAMPLE

CODED BY CLAUDE ACHIM

EMME

TRANSIT LINK CODING FORM

DATE 9-17-76 PAGE 1 OF 1

	ORIGIN NODE	DESTN NODE	S (N)	TIME	REVERSE TIME	USER INFORMATION
1	11	101 *		2.8		
2	12	104 *		10.4		
3	20	105 *		3.4		
4	55	103 *		4.8		
5	56	105 *		4.0		
6	102	103 *		2.5		
7	103	104 *		6.2		
8	103	110 *		3.4		
9	104	105 *		4.4		
10	104	112 *		3.4		
11	105	106 *		6.6		
12						
13						
14						
15						
16						
17						
18						
19						
20						
21						
22						
23						
24						
25						

emme-52-0

27-7-76

EMME

PROJECT EXAMPLE

CODED BY CLAUDE ACHIM

TRANSIT ACCESS LINK CODING FORM

DATE 9-12-76 PAGE 1 OF 1

	Dir	ZONE CENTROID	ACCESS NODE	S (ac)	TIME TO NODE	TIME TO CENTROID	USER INFORMATION
1		1	101 *		20		
2		1	102 *		12		
3		2	11 *		15		
4		2	12 *		24		
5		3	12 *		12		
6		3	13 *		10		
7		4	15		15	18	
8		4	106 *		15		
9		5	31 *		7		
10		5	32 *		5		
11		6	33 *		5		
12		6	34 *		5		
13		6	112 *		8		
14		7	32		8	6	
15		7	54 *		6		
16		8	33 *		10		
17		8	52 *		10		
18		9	126 *		20		
19		9	46 *		15		
20							
21							
22							
23							
24							
25							

emme-53-0

27-7-76

PROJECT ..EXAMPLE.....

CODED BY ..CLAUDE ACHIM.....

EMME

TRANSIT PATH
MULTIPLIER
CODING FORM

DATE ..9-17-76.... PAGE .1.. OF .1...

	mod	ZONE GROUP 'ALL'				ACCESS W1	WAIT W2								USER INFORMATION
	1	2 3 4 5	6 7 8 9 10 11	12 13 14 15 16 17 18	19 20 21 22 23 24 25	26 27 28 29 30 31 32	33 34 35 36 37 38 39	40 41 42 43 44 45 46	47 48 49 50 51 52 53	54 55 56 57 58 59 60	61 62 63 64 65 66 67	68 69 70 71 72	73 74 75 76 77 78 79 80
1		E001		0.5									
2		E002		0.5	1.5								
3													
4													
5													
6													
7													
8													
9													
10													
11													
12													
13													
14													
15													
16													
17													
18													
19													
20													
21													
22													
23													
24													
25													

emme-34-0

24-11-76

EMME

PROJECT ...EXAMPLE.....

CODED BY ...CLAUDE...ACHIM...

ROAD LINK CODING FORM

DATE ..9-17-76..... PAGE ..1.. OF ..3...

	Add Dir	ORIGIN NODE	DESTN NODE	S (+)	TYPE	LENGTH	OBS. TIME	OBS. VOLUME	DATA 1	DATA 2	DATA 3	DATA 4	DATA 5	:	TEXT	USER INFORMATION
	1 2 3 4 5	6 7 8 9 10	11 12	13 14 15	16 17 18 19 20	21 22 23 24 25	26 27 28 29 30	31 32 33 34 35	36 37 38 39 40	41 42 43 44 45	46 47 48 49 50	51 52 53 54 55	56	57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72	73 74 75 76 77 78 79 80
1		12	55*		1.05								:		
2		12	24*		2.6								:		
3		13	56*		1.15								:		
4		14	16*		1.6								:		
5		15	16*		1.25								:		
6		16	17*		1.50								:		
7		16	18*		1.85								:		
8		17	18*		1.05								:		
9		17	43*		2.4								:		
10		18	19*		1.5								:		
11		18	42*		1.7								:		
12		19	20*		0.7								:		
13		19	41*		1.4								:		
14		20	21*		0.8								:		
15		20	38*		1.4								:		
16		20	56*		1.8								:		
17		21	22*		0.75								:		
18		21	36*		1.4								:		
19		22	23*		0.7								:		
20		22	35*		1.40								:		
21		23	24*		0								:		
22		23	25*		0								:		
23		23	26*		0								:		
24		24	25*		0								:		
25		24	26*		0								:		

emme-60-0

27-7-76

EMME

PROJECT ...EXAMPLE.....

CODED BY ...CLAUDE ACHIM..

DEMAND FUNCTION TRAVEL TIME
COEFFICIENT CODING FORM

DATE .9-17-76... PAGE .1.. OF ..1..

	ZONE GROUP 'ALL'	CONSTANT k0	AUTO k1	TRANSIT k2						USER INFORMATION
1	ALL	4.51	0.0521	-0.0521						
2										
3										
4										
5										
6										
7										
8										
9										
10										
11										
12										
13										
14										
15										
16										
17										
18										
19										
20										
21										
22										
23										
24										
25										

emme-70-0

24-11-76

PROJECT ..EXAMPLE.....

CODED BY ..CLAUDE ACHIM.....

EMME

SOCIO-ECONOMIC VARIABLE/
DEMAND FUNCTION COEFFICIENT CODING FORM 2

DATE .....9-17-76 PAGE .1.. OF .1...

a d m	ZONE / VAR. NAME	DESTN ZONE / ORIGIN ZONE	VAR. NAME / ZONE	VALUE	VAR. NAME / ZONE	VALUE	VAR. NAME / ZONE	VALUE	VAR. NAME / ZONE	VALUE	VAR. NAME / ZONE	VALUE	VAR. NAME / ZONE	VALUE	USER INFORMATION
1	1		CAR=	0.8											
2	2		CAR=	1.6											
3	3		CAR=	0.9											
4	ACS	1	2	0.15	3	0.44	5	0.33							
5	ACS	3	1	0.44	2	0.39	4	0.19	5	0.45					
6															
7															
8															
9															
10															
11															
12															
13															
14															
15															
16															
17															
18															
19															
20															
21															
22															
23															
24															
25															

emme-71-0

24-11-76

PROJECT EXAMPLE

CODED BY CLAUDE ACHIM

EMME

SOCIO-ECONOMIC VARIABLE / DEMAND FUNCTION COEFFICIENT CODING FORM 1

DATE 9-17-76 PAGE 1 OF 1

Add'n	ZONE GROUP 'ALL'	DESTN ZONE GROUP	1	2	3	4	5	6	7	8	USER INFORMATION
		KEYS	DCAR ACS	DPSP							
	ZONE ROW	KEYS									
	E001			1.0							
	2			1.0							
	4			1.0							
	6			0.8							
	7			0.78							
	8			0.6							
	9			0.5							

emme-72-0

24-11-76

Appendix B

EMME Command, option and subcommand summary

COMMANDS

Command	Function
$ASSIGN	Perform assignment according to options and subcommands.
$COMMENT	Print comment line.
$DATABANK	Perform operations on the data bank according to subcommands, headers and data.
$DEBUG	Control debug output.
$END	Terminate EMME.
$REPORT	Print reports according to subcommands.
$TITLE	Insert text from command into title line for reports.

$ASSIGN Command options, subcommands and parameters

Option	Function
BIMODAL	Perform bimodal assignment.
ROAD	Perform road traffic assignment (fixed or variable).
TRANSIT	Perform transit assignment (fixed or variable).
FIXED	Fixed demand assignment (road or transit).
VARIABLE	Variable demand assignment (road or transit).

Subcommand	Function
IDENT	Enter text to identify assignment
PARAMETERS	Set assignment parameters.

Assignment parameters are shown in Table B-1.

$DATABANK options, subcommands, headers

Option	Function
NEW	Create a new databank

Subcommand	Function
CORRECTION	Apply permanent corrections (changes) to databank
SCENARIO	Enter scenario data into the databank
USE	Apply temporary changes to databank as specified in scenario data
PRINT	Print data from databank
PUNCH	Punch data from databank
PURGE	Delete data from databank

$DATABANK headers are shown in Table B-2.

$DEBUG command options

Option	Function
YES	Print debugging output (Dumps, messages)
NO	Suppress debugging output.

$END command option

Option	Function
KEEP	Active scenarios are not removed before closing the databank

$REPORT command options, subcommands and parameters

All options, subcommands and parameters listed under $ASSIGN are recognized plus the following.

Subcommand	Function
LINK	Print road link report
DEMAND	Print predicted demand
AUTO TIME DISTRIBUTION	Print automobile trip time distribution.

LINK Parameters:

Parameter	Values accepted	Meaning
CORR	LINK	Corridor specified by a list of links
	NODE	Corridor specified by a list of nodes

If no parameter is specified, all road links are printed.

$TITLE options

Option	Function
1	Fill title line 1 with specified text.
2	Fill title line 2 with specified text.

Table B-1 - Assignment parameters

Parameter	Default	Values accepted	Meaning
AOCC	1.00	x.xx	Auto occupancy global value is x.xx
		PERS	Auto occupancy is matrix from data bank
BCAP	60.	xxx.	Bus capacity is xxx. persons
BEQU	3.00	x.xx	Auto equivalent of 1 bus is x.xx
BUSF	NO[1] or COMP	NO	Bus frequency on road links should not be considered.
		COMP	Bus frequency on road links computed from transit line specifications.
		OBSx	Bus frequency is observed value in user data x for each link.
CONT	NO	YES	Continue road traffic assignment (Not implemented).
		NO	
DFNC	LOGI	LOGI	Logit demand function
		LINE	Linear demand function
DSPD	10.0	xx.x	Default transit speed is xx.x m.p.h.
DTIM	15.0	xx.x	Default transit travel time is xx.x minutes
EXPF	0.75	x.xx	Express bus time factor is x.xx
HDWY	(2)	xxx., xxx.,...	Permissible transit headways are xxx.,xxx.... (1 to 20 values sorted in ascending order).
ITCK	--	xxxx	Bimodal iterations between checkpoints (Not implemented).
ITCP	0	xxxx	Bimodal or transit iterations on automatic capacity (headway) modification
ITLT	0 or 1[3]	xxxx	Bimodal iterations on transit link time modification
ITRD	15	xxxx	Iteration limit within road traffic assignment is xxxx

Table B-1 - Assignment parameters (cont'd)

Parameter	Default	Values accepted	Meaning
LTIM	OBS		Road link times used to calculate transit travel times are:
		OBS	Observed link travel times, if any.
		PRED	Predicted by a road traffic assignment
ODAU	OLD	NEW	Status of automobile demand matrix (not
		OLD	implemented)
ODMA	1	x	Number of OD matrix read from data bank
		name	Name of OD matrix read from data bank
PMUL	NO	YES	Use transit path multipliers
		NO	
RTIM	0	xxxx	Maximum time for a road traffic assignment
SPLT	1.0[4]		Modal split is:
	or	x.xx	global value for percentage of trips by auto
	0.0	AUTO	Matrix of number of vehicle trips read from data bank (MODAL SPLIT) (Not applicable for fixed transit assignment).
		PCNT	Matrix of percentage of trips by auto read from data bank (MODAL SPLIT).
		PERS	Matrix of number of person trips by auto read from data bank (MODAL SPLIT)
		PRED	Matrix of number of person trips by auto predicted by a road traffic assignment (GPQAU).
TDIV	NO	YES	Use line diversion algorithm in transit
		NO	assignment
TIMP	COMP		Transit impedance to be:
		COMP	Computed from transit line specifications
		PRED	Read from previously calculated file.
TOLT	.05	.xxx	Significant relative difference when comparing transit link times.
TOLU	.05	.xxx	Significant relative difference when comparing transit impedances.

Table B-1 - Assignment parameters (cont'd)

Parameter	Default	Values accepted	Meaning
TOLV	.05	.xxx	Significant relative difference when comparing auto time and impedance.
TVAL	YES	YES	Transit network is validated on input.
		NO	Transit network is not validated on input.
VADL	50	xxxx	Variable demand lower limit is xxxx persons. (Total demand less than xxxx is always fixed with specified modal split applied).
VEAD	NO	YES	Use vehicle adjustment matrix in road traffic assignment
		NO	
WAIT	1.0	x.xx	Transit wait time coefficient.
WALK	1.0	x.xx	Transit walk time coefficient.
WFAC	.5	.xxx	Transit regularity factor. Waiting time is .xxx times headway.
WMAX	120.	xxx.	Maximum transit waiting time is xxx. minutes.
WMIN	0.	xxx.	Minimum transit waiting time is xxx. minutes.
WPEN	0.0	xx.x	Transit waiting penalty is xx.x minutes.

(1) COMP for bimodal assignment, NO otherwise.

(2) Default values are 2.,5.,10.,15.,20.,30.

(3) 1 for bimodal assignment, 0 otherwise.

(4) 0.00 for transit-only, fixed demand assignment, 1.00 otherwise.

HEADER	CODING FORM	
	no	name
AUTO OCCUPANCY	03	od matrix
CENTROIDS	01	Centroid
DEMAND FUNCTION COEFFICIENTS	70	demand function travel time coefficient
LINES	50	transit line
MODAL SPLIT	03	od matrix
NODE COORDINATES	02	node coordinate
NODE EXPANSION	51	transit node - road nodes correspondence
O-D MATRIX n aaaa	03	od matrix
ROAD LINKS	60	road link
SCALE x.xx	--	
SOCIONOMIC COEFFICIENTS FORMAT 1	72	socio-economic variable/coeff. form 1
SOCIONOMIC COEFFICIENTS FORMAT 2	71	socio-economic variable/coeff. form 2
SOCIONOMIC VARIABLES FORMAT 1	72	socio-economic variable/coeff. form 1
SOCIONOMIC VARIABLES FORMAT 2	71	socio-economic variable/coeff. form 2
TRANSIT ACCESS LINKS	53	transit access link
TRANSIT LINKS	52	transit link
TRANSIT PATH MULTIPLIERS	54	transit path multiplier
VEHICLE ADJUSTMENT MATRIX	03	od matrix
ZONE GROUPS	04	zone group

Table B-2 - Headers and coding forms